

- 1 -



METHOD, APPARATUS, AND COMPUTER READABLE  
MEDIUM FOR MANAGING MULTIPLE SYSTEM

BACKGROUND OF THE INVENTION

In present-day society, computer systems have been indispensable for providing the living basis supporting our life. Such a computer system as above 5 is required to continue service for 24 hours without stopping. Available as one of these computer systems is an on-line system employed in banks and the like which handles database business affairs as a key or core process. The database business affairs are 10 updated frequently and are therefore permitted of no complete or thorough stoppage. But, on the other hand, there is a demand for creating backups in expectation of protection of data to be handled.

In the database, data to be handled are 15 stored in volumes (hereinafter abbreviated as VOL's) representing memory areas or storages areas precedently set in a disk device so constructed as to include a storage medium such as a magnetic disk and the data are processed. The VOL is a unit of the memory area and is 20 sometimes called by the name of partition. The VOL is identified with a physical VOL identifier (PVID) and the computer system recognizes the VOL by acquiring VOL information through the use of the PVID. The PVID and VOL information are acquired by a disk management

program and saved in a disk management information buffer on an operating system (OS). It will be appreciated that a variety of PVID's are available and in an OS called AIX (registered trademark of IBM Inc.), 5 for instance, the PVID is handled in the form of hexadecimal data such as "005247772d2f36b". An application such as database and the OS accesses (reads and writes) a VOL recognized on the basis of information in the buffer. Let us consider an instance where, 10 for example, an application program accesses a file. In this case, on the basis of a physical device name corresponding to the file to be accessed (hereinafter also termed a physical name and as an example of concreted data, "hdisk0" is used in, for example, the 15 OS called AIX (registered trademark of IBM Inc.), identification information (physical address such as LUN) in a disk device of a physical memory medium identified (from the OS) by that physical device name is designated and a request for access to the disk 20 device is transmitted. At that time, verification is carried out as to whether a PVID stored in a given area of the memory medium to be accessed coincides with a PVID stored in the aforementioned disk management information buffer. If the result of verification 25 shows non-coincidence between the PVID's, this result is handled as an access error.

As will be seen from the above, a technique of realizing a replication of VOL without causing the

database to stop thoroughly is important. The prevention of complete stoppage intends to attain recovery of a system within a short period of time in the range of not causing troubles in business affairs

5 even in the event that the system becomes faulty so that system users may be prevented from recognizing that the system stands stopped. To this end, a technique disclosed in USP6401178 (corresponding to JP-A-2002-41368) is known according to which replica of

10 data in a disk is executed in a disk device and this technique is applied to VOL's to provide a technique called VOL replica. The VOL replica includes two means for pair construction and pair division directed to paired VOL's of original VOL representing a data

15 replication source and copy VOL representing a data replication target. The pair construction is means for fast creating a copy VOL representing a replica of an original VOL by making coincident (synchronized) all data pieces including PVID's of original/copy VOL's and

20 VOL information. Accordingly, in a status of pair construction, the PVID's of the original/copy VOL's are coincident with each other, thereby providing a function of enabling the host computer system to recognize the original/copy VOL's impersonating a sole

25 VOL. On the other hand, the pair division is means for cancelling the synchronization of the original/copy VOL's and rewriting the PVID of the copy VOL of the paired VOL's taking the pair construction into a PVID

different from that of the original VOL. This provides a function of dividing the paired VOL's recognized as a sole VOL by the host computer system during pair construction and enabling the system to recognize the 5 original and copy VOL's as different VOL's. These two means provide a function of creating a replica of the original VOL at a high speed and permitting the computer system to operate a replicated copy VOL.

On the other hand, a computer system required 10 to have high reliability of recovery within short period of time is so constructed as to include an active computer or a living computer for executing processes and a standby computer for taking over a process in the event that a fault occurs in the living 15 party. A cluster program for managing the living and standby parties provides a procedure for handing the process to the standby party at the time that the fault occurring in the living party is detected. For handing over the process, data used in the application and OS 20 must be handed over. For example, in the aforementioned database system, it becomes necessary that information concerning a VOL in which data to be handled is stored be handed over.

#### SUMMARY OF THE INVENTION

25 In the aforementioned VOL replication techniques, however, replication is carried out inside the disk device and therefore any other computer than

that having executed the replication has no information concerning replicated data. Accordingly, in the event that a fault occurs in a living party computer having executed replication and the party switchover takes

5 place in a computer system applied with the cluster program, a PVID of a volume which is subjected to the aforementioned replication and which is stored in the disk management information buffer of a standby party having taken over the process is conditioned not to

10 coincide with a PVID stored in a memory medium inside the disk device and corresponding to the volume of interest. When the standby party computer accesses that volume under this condition, the result of verification of PVID's shows non-coincidence and an

15 access error results. The aforementioned prior arts fail to consider these points.

More specifically, in case a fault takes place in the living party/standby party computer system sharing original/copy VOL's subjected to VOL replication (pair construction and pair division), the standby party fails to access a copy VOL, then the standby party cannot take over the living party. This is because the PVID of the copy VOL is rewritten through the VOL replica and this change is reflected upon the

20 living party whereas the standby party tries to access the copy VOL in accordance with information stored in the disk management information buffer before the

25 execution of the VOL replica.

As will be seen from the above, when utilizing the two means of cluster program and VOL replica for attaining high reliability, the conventional techniques raise a problem that a 5 situation occurs in which a process undertaken by both the living and standby parties cannot be taken over.

A first object of the present invention is to provide method and system which can reflect a change of a copy VOL caused by both the VOL replica means upon 10 the standby party.

A second object of the invention is to provide method and system in which, when a fault occurs in the living or standby party in the course of the fact that a copy VOL is changed by VOL replica means 15 and reflection of the change is executed, the change of the copy VOL and the reflection of the change can be taken over.

A third object of the invention is to provide method and system in which, when a fault occurs in the 20 living party after execution of the VOL replica means, a process having been executed by the living party can be taken over to the standby party.

A fourth object of the invention is to provide method and system in which, when a standby 25 party in association with a living party using a computer system utilizing the VOL replica means is newly added, a change of a copy VOL can be reflected upon the standby party.

A fifth object of the invention is to provide method and system in which, when a standby party in association with a living party using a computer system utilizing the VOL replica means is newly added and a 5 fault occurs in the living party or in the standby party in the course of reflection of a change of a copy VOL upon the standby party, the change of the copy VOL and the reflection of the change can be taken over.

A sixth object of the invention is to provide 10 method and system in which, when a standby party in association with a living party using a computer system utilizing the VOL replica means is newly added and thereafter a fault occurs in the living party, a process executed by the living party can be taken over 15 to the standby party.

According to the present invention, in a high available computer system comprised of a living party/standby party computer system and in which the living/standby parties share paired VOL's subjected to 20 execution of the VOL replica, a physical name of a copy VOL to be changed by the VOL replica is acquired during start of a cluster program. For example, by reading a physical name of a copy VOL saved in a file for setting the VOL replica, the physical name can be acquired.

25 Further, when executing the VOL replica by means of the living party, the living party informs the standby party of the time of start of execution of the VOL replica and after completion of the VOL replica,

thereby enabling the standby party to recognize that the living party has changed the copy VOL.

When being informed of the change of the copy VOL, the standby party conducts a process for reflecting the change in copy VOL status. For example, when the pair division is carried out in the VOL replica, a physical name of a copy VOL which has already been acquired is used to acquire the PVID of the copy VOL set newly, and further, information of the copy VOL is acquired using that PVID. In this manner, copy VOL information after the change is reflected to permit the standby party to access the copy VOL.

When completing the reflection of the copy VOL information, the standby party informs the living party that the reflection is completed. By receiving this notice, the living party recognizes that consistency of the copy VOL information is guaranteed.

Each of such a plurality of computer parties has, as a status flag, a VOL replica status indicating whether the VOL replica has been executed in the living party and whether the copy VOL information is reflected upon the standby party. The status flag is also stored, as a status file, on the computer system. Further, when interchanging information between the living and standby parties, the status flag is informed to the partner side in order that process states of the living and standby parties can be recognized.

In addition, a VOL replica status at the time

of starting the cluster program is also acquired to examine whether the copy VOL information is coincident between the living and standby parties. For example, by reading a copy VOL reflection status saved in the 5 VOL replica status file, it is decided whether the copy VOL information is reflected upon both the living and standby parties. In case the copy VOL information is not reflected upon the standby party, the living party commands the standby party to reflect the copy VOL 10 information thereupon and the standby party fulfills the reflection of the copy VOL information. In this manner, even when the VOL replica process or the copy VOL information reflection process is interrupted, the interrupted process can be taken over and resumed.

15 In describing the present invention, the term PVID is used but it is to be noted that the "volume" to be identified with this identifier may be managed in a unit of any type. For example, in connection with the "volume", a unit termed LU (logical unit), a unit 20 obtained by somewhat dividing the LU or a unit constructed of several LU's may be handled as "volume".

#### BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a high-level system block diagram showing a living/standby computer system model 25 according to an embodiment of the invention.

Fig. 2 is a high-level block diagram showing a conventional fault take-over system using a

living/standby computer system model.

Fig. 3 is a low-level block diagram showing the computer system according to the embodiment of the invention.

5           Fig. 4 is a flowchart illustrating the outline of procedures in the living/standby party computers when the living party computer carries out pair division of volume replica process in the computer system according to the embodiment of the invention.

10          Fig. 5 is a flowchart illustrating the outline of procedures in the living/standby party computers when the living party computer carries out pair reconstruction of volume replica process in the computer system according to the embodiment of the

15 invention.

Fig. 6 is a flowchart of a pre-process in the living/standby party computers.

Fig. 7 is a flowchart of a fault monitor and party switchover process in the living/standby party computers.

20          Fig. 8 is a flowchart in the living/standby party computers illustrating procedures for the living party computer to execute and complete the volume replica process.

25          Fig. 9 is a flowchart in the living/standby party computers illustrating procedures for the standby party computer to execute and complete a process of reflecting changed copy volume information.

Fig. 10 is a flowchart illustrating procedures for returning to the fault monitor and party switchover process after the reflection of the copy volume information in the living/standby party 5 computers.

Fig. 11 is a flowchart illustrating procedures for the living party computer to take over the copy volume replica process and copy volume in the event of the occurrence of a fault.

10 Fig. 12 is a flowchart illustrating procedure for the living party computer to hand over to the standby party computer the volume information taken over by the living party computer during the occurrence of a fault.

15 Fig. 13 is a table showing an example of information contained in a VOL replica definition file.

Fig. 14 is a table showing an example of information contained in a disk management information buffer.

20 Fig. 15 is a table showing an example of information held by a volume management section.

Fig. 16 is a table showing an example of a replica status management table representing one type of management form of the VOL replica status flag.

## 25 DETAILED DESCRIPTION OF THE EMBODIMENTS

It should be understood that drawings and description illustrative of the present invention are

simplified to show appropriate components for better and clear understanding of the invention and known components are omitted. In connection with the technique of the present invention, some other 5 components of the conventional techniques seem to be desirable and/or necessary for carrying out the present invention. But these components in the conventional techniques are known and are not effective to make the present invention understandable easily and will not be 10 described herein. The present invention will now be described in greater detail with reference to the accompanying drawings.

The present embodiment intends to provide a VOL information consistency guaranty system capable of 15 reflecting information of a VOL representing an object of VOL replica executed by a living party computer upon a standby party computer.

Illustrated in Fig. 1 is a block diagram of a living/standby party computer model according to the 20 present embodiment.

Illustrated in Fig. 2 is a block diagram of a living/standby party computer model based on the conventional technique and having problems to be solved by the present embodiment.

25 Each of the models as shown in Figs. 1 and 2 comprises a computer layer for performing processes and a disk layer for saving data necessary for the processes. The computer layer includes a plurality of

living party computers 10 and a plurality of standby party computers 20. Each computer has means 01 for mutual communication. Each of the computers 10 and 20 includes four programs as below: that is,

5           (1) Operating systems (OS's) 11 and 21 for controlling operation of the computers,

              (2) Disk management programs 12 and 22 for performing disk management,

              (3) Cluster programs 13 and 23 for realizing a  
10 highly utilizable system based on party switchover, and

              (4) Applications 14 and 24.

Each of the cluster programs 13 and 23 has a function of interchanging mutual information and a fault monitoring function by using the communication means 01. On the other hand, the disk layer includes a disk 30 shared by the host computers. The disk 30 is comprised of two components, that is, (1) VOL's 32 and 35 for saving information and (2) a VOL management mechanism 31 for controlling VOL's in the disk device.  
15 The VOL's 32 and 35 have PVID's 33 and 36, respectively, for identifying objects to be accessed from the host computer layers 10 and 20 and VOL information pieces 34 and 37, respectively.

In the technique shown in Fig. 2, (1) a  
25 change of copy VOL information on disk due to a VOL replica carried out with the disk 30 in a process by the living party 10 is not reflected upon the standby party 20 and therefore, (2) when a fault occurs in the

living party 10 and (3) party switchover is effected by means of the cluster programs 13 and 23, however, (4) access to a copy VOL 35 fails. Consequently, the process applied by the application 14 on the living 5 party 10 to the copy VOL 35 cannot sometimes be taken over normally by the application 24 on the standby party 20.

In Fig. 1, by taking the opportunity of the fact that (1) change of copy VOL information on disk is 10 executed by the disk 30 through VOL replica in a process of the living party 10, (2) the executed change is informed from the living party cluster program 13 to the standby party cluster program 23 by way of the communication means 01. Through this, (3) the right to 15 control the copy VOL 35 is temporarily switched from the living party 10 to the standby party 20 and (3) thus changed copy VOL information pieces 36 and 37 are reflected upon the standby party 20. After the reflection, the living party 10 regains the right to 20 control the copy VOL 35 and executes a process applied to the copy VOL 35. Since this guarantees consistency of the copy VOL information between the living and standby party computers, the process applied to the copy VOL can be taken over in the event that a fault 25 occurs subsequently.

Referring to Fig. 3, there is illustrated, in simplified block diagram form, the living/standby party computer system according to the present embodiment.

Typically, the system of Fig. 3 comprises two components, that is, (1) a computer layer including a plurality of application computers (corresponding to 10 and 20 described previously) and (2) a disk layer for saving data shared by the computer layer (corresponding to 30 described previously). In Fig. 3, for clarity of description, individual programs are labeled by numerals of three figures. In numbering, a numeral of the same two lower figures is used for the same program in the living and standby party computers and the location of hundred is "1" for designating the living party computer and "2" for designating the standby party computer. In the following, individual programs will first be described. In description, the programs of the respective computers are described by making reference to only program numbers on the living party computer with the intention of also giving a description of the corresponding programs on the standby party computer.

A disk 300 includes a volume management section 310 and original and copy VOL's 320 and 330 subjected to VOL replica, the original and copy VOL's (320 and 330) having PVID's (321 and 331) for identification of these VOL's, respectively, and VOL information pieces (322 and 332) necessary for access to the VOL's, respectively.

The volume management section 310 functions to execute the VOL replica and change the PVID's (321

and 331) and VOL information pieces (322 and 332) of original VOL 320 and copy VOL 330. Management information the volume management section 310 has is shown in, for example, Fig. 15. For original and copy 5 VOL's 1501 and 1511, the management section 310 holds storage positions 1502 and 1512 of PVID's and storage positions 1503 and 1513 of VOL information pieces while mutually associating the individual storage positions. When a request is made to the volume management section 10 for accessing the PVID's and VOL information pieces of the original/copy VOL's, the PVID's and VOL information pieces are read out of the respective corresponding storage positions to respond to the request. Here, the information held in the management section is set to 15 the storage positions of PVID's and VOL information pieces but the PVID's and VOL information pieces per se may be held. The VOL information may include identifiers for identification of volumes in addition to the PVID's. Alternatively, the information held in 20 the management section may be held in a disk management program 150 or disk management information buffer 131, thereby ensuring that the management program 150 can change the PVID's and VOL information pieces of original/copy VOL's through the medium of the volume 25 management section 310.

A living party computer 100 includes an OS 130, a cluster program 120, the disk management program 150, an application 110, a VOL replica definition file

160 and a VOL replica status file 140.

The OS 130 includes the disk management information buffer 131. The OS 130 also intervenes in access from the disk management program 150 to the disk 5 300. In this access, the result of access is sometimes saved in the disk management information buffer 131 and in some applications, the information saved in the buffer 131 is utilized without accessing the disk 300. The pieces of information held by the disk management 10 information buffers 131 and 231 are shown in, for example, Fig. 14. Each of the buffers 131 and 231 includes physical names 1401, 1411 and 1421 of VOL's, PVID's 1402, 1412 and 1422 of the VOL's and VOL 15 information pieces 1403, 1413 and 1423 of the VOL's. Physical name 1401 of a certain VOL 1 makes the correspondence with VOL1 PVID information 1402 indicative of a PVID of the VOL1 and with VOL 20 information 1403 of the VOL1. Physical names of other VOL's also make the correspondence with corresponding PVID's and VOL information pieces.

The VOL replica definition file 160 includes definitions necessary for execution of a VOL replica and is constructed of a table as shown in, for example, Fig. 13, including physical names of the original VOL 25 320 and copy VOL 330 subjected to replica. The replica is applied to an original VOL and a copy VOL to be paired with each other and in Fig. 13, physical names of original VOL's and copy VOL's constituting

individual pairs are stored while making the correspondence between original and copy VOL's in the individual pairs. For example, in Fig. 13, original VOL 1601 and copy VOL 1602 are paired as indicated.

5           The disk management program 150 has programs for accessing the disk 300 executable on the living party computer 100, for example, including a program for lock control of VOL's, a program for acquisition of PVID's and VOL information of VOL's and a VOL replica  
10 execution program. During execution of these programs, the program 150 sometimes commands the volume management section 310 or reads the disk management information buffer 131. Further, the VOL replica execution program sometimes utilizes the VOL replica  
15 definition file 160.

          The cluster program 120 has a copy VOL identification information buffer 121 adapted to save information for identifying copy VOL's subjected to a VOL replica, a VOL replica status buffer 122 adapted to  
20 hold the execution status of the VOL replica, a communication section 123 adapted to make communication with the other party, a monitor section 124 adapted to provide a function of monitoring states of the self and other parties, and a switchover section 125 adapted to  
25 perform a process concerning the party switchover. The copy VOL identification information buffer 121 is a buffer for holding identification information of a copy VOL read out of the VOL replica definition file 160 and

is constructed of the table shown in Fig. 13, including physical names of the original and copy VOL's 320 and 330 subjected to the replica.

The monitor section 124 has a function of  
5 detecting a fault of the party of its own and the execution of an MRCF (replica creating process) by monitoring the application 110, a function of informing a state of the self-party by communicating with the communication section 223 of cluster program 220 of the  
10 standby party through the medium of the communication section 123 and a function of detecting a faulty state of the other party and a VOL replica status.

The party switchover section 125 has a party switchover function for performing switching between  
15 the living party and standby party in accordance with faults in the self and other parties detected from the monitor section 124. The switchover section 125 also has a function to respond to detection of VOL replica execution statuses of the self and other parties from  
20 the monitor section 124 so as to control execution of the VOL replica through the medium of the disk management program 150, a function to hold the statuses in the VOL replica status flag 122 and VOL replica status file 140, and a function to inform the  
25 application 110 utilizing the VOL replica that the use of the copy VOL is to be stopped/resumed. Further, the switchover section 125 also has a function to read the VOL replica definition file 160 and hold information

necessary for identifying a copy VOL 330 in the copy VOL identification information buffer 121.

It is to be noted that a computer of the living party can function to fulfill, for example, the 5 party switchover section by executing a predetermined program in the computer.

The program for making the living party, standby party and disk device function as the party switchover section or the like is recorded on a 10 recording medium such as CD-ROM and stored in a magnetic disk, for instance, and thereafter loaded on the memory so as to be executed. The recording medium for recording the program may be other recording media than the CD-ROM. In an alternative, the program may be 15 installed from the recording medium to the information processing apparatus and then used or may be used by accessing the recording medium through a network.

Fig. 4 and ensuing figures illustrate flows of processes. For avoidance of confusion with numerals 20 in Fig. 3, numerals of four figures are used in each drawing. Reference numerals in Figs. 4 to 12 have each two upper figures corresponding to the figure number and two lower figures including 01 to 20 indicative of process steps on the living party computer, 21 to 40 25 indicative of process steps on the standby party computer and 41 to 60 indicative of process steps on the disk device. Further, data interchange process steps carried out between each computer party and the

disk device are designated by two lower figures 80 to 99. It will be appreciated that in the following description, even when a description is given by way of the process steps in the living party, corresponding 5 process steps in the standby party will sometimes be carried out similarly unless noted specifically.

Illustrated in Figs. 4 and 5 are simplified flowcharts of processes in the living/standby party computer model according to the present embodiment, 10 with Fig. 4 indicating an instance where the pair division of VOL replica means is executed and Fig. 5 indicating an instance where the pair reconstruction of VOL replica means is executed.

In Fig. 4 or 5, the process flow is divided 15 into two major phases, that is, (1) a pre-process phase in which information necessary for performing the copy VOL information reflection process is processed before execution of VOL replica and (2) a copy VOL information consistency guaranty process phase including execution 20 of the VOL replica means in the living party. Details of each phase will be described sequentially by making the correspondence with the system block diagram of Fig. 3.

The pre-process is common to Figs. 4 and 5 25 which the living and standby party computers first carry out in common pre-process. In the pre-process, a VOL replica definition is first read (0401). This includes a step in which the cluster program 130 on the

living party computer 100 reads the VOL replica definition file 160. From this definition file, a physical name of a copy VOL subjected to a VOL replica is acquired (copy VOL physical name acquisition step 5 0402). Up to here, the pre-process ends and the living party 100 executes a process not applied with the present embodiment until the copy VOL changes, thus typically continuing to a normal living state 0403.

Subsequently, when the copy VOL change is 10 started, the aforementioned consistency guaranty process phase is executed. The consistency guaranty phase includes three stages in total, that is, (1) stage X representing a copy VOL change stage for executing the VOL replica with the living party, (2) 15 stage Y representing a copy VOL change reflection stage for reflecting copy VOL information changed in the stage X upon the standby party, and (3) stage Z representing a copy VOL working resumption stage for resuming working of the copy VOL.

20 The pre-process is common to Figs. 4 and 5 but in the copy VOL information consistency guaranty phase, the contents of change of the copy VOL information in stage X differs for Figs. 4 and 5 and besides, in the copy VOL information reflection process 25 of stages Y and Z, accesses to the copy VOL and lock of copy VOL are needed. Therefore, different steps are carried out in Figs. 4 and 5.

The processing flow in the individual stages

will now be described in sequence.

In the stage X, when a change of a copy VOL is first executed by means of the disk management program 150, this change is informed to the switchover section 125 on the cluster program 120 (copy VOL change notice 0404). By receiving this notice, the party switchover section 125 informs the cluster program 220 of standby party 200 that the copy VOL change is executed, through the medium of monitor section 124 and communication section 123 (rightward arrow 0581). The standby computer 200 receives the notice 0581 at the switchover section 225 through the medium of the communication section 223 and monitor section 224 on the cluster program 220 and recognizes that the copy VOL change process is executed in the living party 100 (copy VOL change start recognition step 0524). After the recognition, the standby party 200 informs the living party 100 of the recognition through a route inverse to the route through which the execution is informed from the living party 100 to the standby party 200 (leftward arrow 0581) and waits for reception of a notice to the effect that the copy VOL change is completed.

Next, when the living party 100 receives, at the party switchover section 125, the recognition notice leftward arrow 0581 from the standby party, it returns the process to the disk management program 130 to execute a step accompanying the copy VOL change (VOL

replica execution 0405). This step is taken over to the volume management section 310 on the disk device 300 through the OS 130 or the disk management buffer 131 on the OS. The management section 310 acquires the  
5 right to control the copy VOL and executes the following steps. Firstly, in the case of pair construction 0541 in Fig. 5, the management section 310 informs the living party computer that the a PVID 331 of the copy VOL 330 is changed to have the same value  
10 as a PVID 321 of the original VOL 330 and informs the living party that the step is completed (arrow 0582). On the other hand, in the case of pair division in Fig. 4, the management section 310 changes, to another unique value, the value of PVID 331 of the copy VOL  
15 made to be equal to the value of PVID 321 of the original VOL by means of the pair construction, changes the VOL information 332 of the copy VOL from the value prevailing before the execution of pair construction and informs the living party computer of information  
20 indicative of end of the step (arrow 0482).

Through the steps in the stage X as above, the execution of the copy VOL change step in the living party can advantageously be informed to the standby party. This brings about an advantage that when a  
25 fault occurs in the living party before the standby party recognizes the copy VOL change completion, the standby party can recognize whether the change step has been executed, thereby permitting the self-party to

recognize a process to be taken over after the occurrence of the fault.

Subsequent to the stage X, the stage Y and ensuing stage are executed, in which steps are  
5 different for the pair division mode (Fig. 4) and the pair construction mode (Fig. 5). The steps will now be detailed in sequence of Figs. 4 and 5. The step of stage Y is initiated by taking the opportunity of the fact that the monitor section of the living party  
10 detects the notice transmitted from the disk device in 0482 (or 0582).

In the case of pair division (Fig. 4), the living party 100 releases the right to control the copy VOL 320 acquired during the VOL replica step 0405 in  
15 order to enable the standby party 200 to execute a process applied to the copy VOL 320 (copy VOL release 0406). In this step, the party switchover section 125 calls, through the management program 130, the management section 310 to enable it to release the  
20 right to control the copy VOL acquired in the VOL replica step 0405 (rightward arrow 0483). The management section 310 releases the right to control the copy VOL 320 (copy VOL release 0442) to complete this step.

25 As the copy VOL release step 0406 ends, a copy VOL change completion notice step 0407 is executed in which the party switchover section 125 of living party 100 informs the party switchover section 225 on

standby party 200 of the completion of the copy VOL change through the medium of a path similar to that used during the notice of execution (rightward arrow 0484). After confirming that the standby party has

5 received the copy VOL change notice, the living party 100 ends the copy VOL change notice step 0407 and waits for the standby party to complete reflection of the changed copy VOL information. It will be appreciated that in the copy VOL change completion notice step

10 0407, the information to be notified may include information (hereinafter referred to as information 1) for identifying the copy VOL whose copy VOL information (such as PVID) is changed in the step 0441. The information 1 can be a physical device name of the copy

15 VOL or an identifier for a pair constituted by the copy VOL in Fig. 13 or information (hereinafter referred to as information 2) for indicating whether the copy VOL change step in 0441 is concomitant with the pair division or pair construction ((or information for

20 identifying whether the reflection step of the copy VOL information to be executed by the standby party in the stage Y is "acquisition of a newly assigned PVID (corresponding to 0427 in Fig. 4)" or "erase of the PVID (corresponding to 0526 in Fig. 5)). Here, the

25 information 1 and information 2 may be transmitted to the standby party computer at the timing different from that for copy VOL change completion notice. In this case, too, by taking the opportunity of detection of

the copy VOL change completion notice 0482 (or 0582) from the disk device or detection of the copy VOL release completion notice (0483) from the disk device, the information 1 and information 2 will be  
5 transmitted.

The notification of the information 1 permits the standby party to recognize for which copy VOL the PVID change step (0427, 0526) is to be executed. Also, the notification of the information 2 makes it possible  
10 to decide which one of the steps 0427 and 0526 is to be executed.

On the other hand, the party switchover section 225 on standby party 200 receiving the copy VOL change notice recognizes that the copy VOL change has  
15 been carried out in the living party (copy VOL change completion recognition step 0425), carries out a step of updating the VOL replica status flag and performs, through the disk management program 230, a step of reflecting the copy VOL information.

20 The step of updating the VOL replica status flag includes storing information corresponding to flag "B2" and information 2 in a table for managing the VOL replica status flag while making the correspondence between the information and a physical device name of  
25 the copy VOL identified by the information 1. Firstly, the right to control the copy VOL is acquired (copy VOL acquisition 0426). The copy VOL acquisition step is carried out in accordance with a processing flow

similar to that of the copy VOL release step (arrow 0485, copy VOL acquisition 0443).

After the copy VOL acquisition step 0426, the copy VOL PVID acquisition step 0427 is carried out. In 5 the PVID acquisition step 0427, the party switchover section 225 uses the copy VOL physical name acquired in the pre-process step 0442 to execute the disk management program 250. The program 250 calls the volume management section 310 on the disk device 300 10 directly without routing through the disk management information buffer 231 on the OS (leftward arrow 0486), and the management section 310 reads a PVID 331 of the copy VOL 330 (copy VOL PVID acquisition 0444) and returns it (rightward arrow 0486). Through this, the 15 party switchover section 225 acquires the PVID 331 of the copy VOL (copy VOL PVID acquisition step 0427). At that time, the disk management program 250 stores that PVID 331 of the copy VOL acquired in the step 0427 in the disk management information buffer 231.

20 After the copy VOL PVID acquisition step 0427, a copy VOL information acquisition step 0428 is carried out. In the VOL information acquisition step 0428, the party switchover section 225 uses the PVID 331 of copy VOL acquired in the PVID acquisition step 25 0427 to execute the disk management program 250. The program 250 calls the management section 310 in a process flow similar to that in the step 0427 to acquire VOL information 332 of the copy VOL (arrow

0487, copy VOL information acquisition step 0445). At that time, too, like the step 0427, the disk management program 250 stores the acquired copy VOL information 332 in the disk management information buffer 231.

5           After completion of the step 0428, the party switchover section 225 performs a step of releasing the copy VOL in the same process flow as that in the copy VOL acquisition step 0426 (copy VOL release steps 0429, 0466 and arrow 0488).

10           On the other hand, in the case of pair construction (Fig. 5), the copy VOL 320 is changed (synchronized) and combined with the original VOL 310 so as to be viewed or recognized as a sole VOL from the living party 100 and standby party 200. Accordingly, 15 it is necessary that the standby party 200 be prevented from accessing the nonexistent (invisible) copy VOL prevailing before the change in accordance with a PVID 331 and VOL information 332 of the copy VOL stored on the disk management information buffer thereby to cause 20 an error. Therefore, the living party 100 informs the standby party 200 of completion of the copy VOL change through a copy VOL change completion notice step 0506 (arrow 0582). In the step 0506, notice (arrow 0582) and copy VOL change completion recognition step 0525 by 25 the standby party, processes comparable to those in the step 0407, notice (arrow 0484) and step 0425 are carried out, thus enabling the party switchover section 225 on the standby party 200 to recognize that the copy

VOL has been changed.

Having recognized the copy VOL change, the party switchover section 225 executes the copy VOL information erase step 0526 representing a step of 5 reflecting the copy VOL information, through the medium of the disk management program 250. In the step 0526, the program 250 applies a process to the disk management information buffer 231 so that the PVID 331 and VOL information 332 of the copy VOL stored on the 10 buffer 231 and placed in the pair division condition may be erased.

In this manner, the copy VOL information reflection stage Y is completed. This brings about an advantage that during the pair division (Fig. 4) in 15 which the copy VOL information changed by the VOL replica executed by the living party can be reflected upon the standby party, the standby party can access the copy VOL and an advantage that during the pair construction (Fig. 5), the standby party can be 20 prevented from accessing the nonexistent (invisible) copy VOL.

The pair construction mode described herein refers to start of replica during which a PVID and VOL information of a copy VOL are changed and the copy VOL 25 is viewed as being concealed from the host layer. Accordingly, only an original VOL is recognizable from the host layer and as a result, access to the original VOL alone is permitted. In the present system,

however, write to the original VOL can be reflected upon the copy VOL synchronously or asynchronously.

The pair division mode described herein refers to end of the replica during which the PVID and 5 VOL information of the copy VOL are changed while the contents of the VOL being kept to be conditioned to be the same as that of the original VOL in the pair division mode and the copy VOL is recognizable, from the host layer, as a single VOL separated from the 10 original VOL. The host layer is conditioned as being permitted to issue access requests to the original VOL and copy VOL separately.

After the stage Y, the stage Z is executed. Following the step 0429, the standby party 200 executes 15 reflection completion notice 0430 and the party switchover section 225 of standby party 200 informs, through a similar path to that used for notifying the execution, the party switchover section 125 on living party 100 that the copy VOL information is reflected 20 upon the standby party (arrow 0489). The standby party 200 confirms that the living party has received the notice and ends the step 0430.

It is to be noted herein that in step 0410, data excepting the PVID and VOL information are 25 sometimes non-coincident between the original and copy VOL's. The reason for this is as below. After the pair division (copy VOL change step 0401) in which data of the original and copy VOL's are coincident with each

other, accessing to the original VOL continues until the step 0410 is executed and as a result, the original VOL will sometimes be updated. In case the coincidence between the original and copy VOL's is needed, update  
5 of the original VOL will temporarily be limited between the copy VOL change step 0401 and the step 0410 or, if the limitation is not imposed, synchronization between the original and copy VOL's will sometimes be made before working of the original and copy VOL's is  
10 started in the step 0410.

On the other hand, the party switchover section 125 of living party receiving the notice (arrow 0489) recognizes that the copy VOL information is reflected in the standby party (reflection completion  
15 recognition step 0408) and obtains the right to control the copy VOL through the disk management program 150 (copy VOL acquisition 0409). In this manner, the stage Z is completed and the living party shifts to the normal working status 0410 to start working both the  
20 original and copy VOL's whereas the standby party shifts to a normal standby status 0431.

On the other hand, during the pair construction (Fig. 5), like the step 0430, information reflection completion notice (arrow 0489) and step  
25 0408, the standby party executes a reflection completion notice step 0527 and an information reflection completion notice 0528 and the living party executes a reflection completion step 0507 in the

living and standby party computers 100 and 200. In this manner, the stage Z is completed and the living party shifts to a normal working status 0508 to continue working of the original VOL whereas the 5 standby party shifts to a normal standby status 0528.

Through a series of steps as above, both the living and standby party computers can advantageously recognize the PVID and VOL information of the copy VOL changed by the VOL replica to guarantee the consistency 10 of information of the copy VOL. Thus, even in the event that a fault occurs since then in the living party, the standby party can access the copy VOL to permit normal take-over of business affairs of both the original and copy VOL's, thereby solving the 15 conventional problems.

Referring now to Figs. 6 to 12, there are illustrated flowcharts showing details of process flows in the present embodiment. In the following description, when both the living and standby party 20 computers engage in processes, the process by the living party is indicated on the left side and the process by the standby party is indicated on the right side. Individual steps in the figures will be described by making the correspondence with the steps 25 in Figs. 3 to 5 but for simplicity of description, the steps appearing in the description given in connection with Figs. 3 to 5 will sometimes be omitted in the following.

Flowcharts as shown in Fig. 6 depict details of the aforementioned pre-processes executed by the living and standby party computers. Since the pre-processes carried out in the living and standby parties 5 are similar to each other, the pre-process by the living party will be described in the following.

Firstly, step 0601 for confirming the presence or absence of a VOL replica definition file is carried out to access the VOL replica definition file 10 160 (arrow 0681). In the absence of the file, the VOL replica process is not executed and therefore any special step need not be done, thus ending the pre-process.

On the other hand, in the presence of the 15 file, the file 160 is read (step 0602, arrow 0682) and step 0603 of acquiring a physical name of a copy VOL subjected to a VOL replica is carried out. Here, the step 0602 corresponds to the aforementioned step 0401 and the step 0603 corresponds to the aforementioned 20 step 0402.

Further, a VOL replica status is read out of the VOL replica status file 140 and is stored in the VOL replica status flag 122 (step 0604, arrow 0683). This is because when the cluster program is restarted 25 in the course of execution of the VOL replica, it is necessary to recognize whether copy VOL information needs to be reflected upon the standby party. Here, the VOL replica status file and VOL replica status flag

can be managed by means of such a table as a replica status management table shown in Fig. 16. In Fig. 16, in column 1600 of pair identifier, values of an identifier for identifying a pair constructed of an  
5 original VOL and a copy VOL are stored. Then, physical device names of the original VOL and copy VOL constituting the pair are stored in column 1601 of original VOL physical name and column 1602 of copy VOL physical name, respectively. Further, flags indicative  
10 of the states in the living and standby party computers related to changes of PVID of copy VOL are stored in column 1603 of living party status flag and column 1604 of standby party status flag, respectively. In addition, information indicating whether the status of  
15 execution of replica applied to each pair is pair division or pair reconstruction is stored in column 1605 of division flag. In Fig. 16, for example, a pair designated by pair identifier "1" indicates that this pair is constructed of an original VOL having a  
20 physical name of "hdisk0" and a copy VOL having a physical name of "hdisk100". Then, it is indicated that the status flags in the living and standby parties are "A1" and "B1", respectively, and the execution status of replica is "pair division" in relation to the  
25 copy VOL "hdisk100".

It is to be noted that Fig. 16 is illustrative of the living party status flag and standby party status flag managed by the same table but

tables may be employed which manage these flags separately. In this case, the replica status management table consists of two kinds of tables of which one eliminates either the column 1603 of living party status flag or the column 1604 of standby party status flag in Fig. 16.

Through the above, the pre-process ends, thus shifting to the normal living state 0403 (7A in Fig. 7).

Illustrated in Fig. 7 is a flowchart indicative of how the conventional fault monitor process and party switchover process based on the cluster program is related to the VOL replica/copy VOL information consistency guaranty process. The process referred to herein corresponds to the normal working status of living party (step 0410 or 0508) and the normal standby status of standby party (step 0431 or 0528) in Fig. 4 or 5.

In Fig. 7, the cluster program 120 on living party 100 first checks the status of the self-party by means of the party switchover section 125 (step 0701) and decides whether the party switchover is necessary (step 0702). The self-party status check step 0701 includes communication made between the cluster program 120 and the application 110 (arrow 0782). The communication 0782 includes information as to whether there occurs an application fault, which requires the party switchover, or a request for execution of VOL

replica from the application.

In case the party switchover is determined to be necessary in the step 0702, the cluster program 120 informs the application 110 that the process is to be 5 interrupted for the purpose of fulfilling the party switchover (arrow 0783) and the process is switched to the standby party by means of the party switchover section 125 (step 0703). Thereafter, since the computer party serving as the former living party 100 10 has been switched to the standby party, the cluster program 120 executes the monitor process in the standby party (7B in Fig. 7).

On the other hand, if the party switchover is not necessary, the party switchover section 125 15 communicates with the party switchover section 225 of standby party 200 through the monitor sections 124 and 224 and the communication sections 123 and 223 (arrow 0781) to communicate the self-party (living party) status and check the status of the other party (standby 20 party) (step 0704). At that time, the communication 0781 includes interchange of information consisting of the VOL replica status flag 122 or the like managed in the format of the replica status management table. The reason for this is as below. By watching the VOL 25 replica statuses of the self and other parties, both the parties can recognize whether the copy VOL status is changed/reflected and therefore, when the party switchover takes place or a standby party is newly

added during the VOL replica, it can be decided whether reflection of the copy VOL information is necessary.

Also, in the event that the other party becomes faulty

and fails to communicate in the communication 0781, by

5 some kind of failures the other party is considered as being conditioned to fail to make a decision and the process proceeds to the following step. It is also to be noted that in the following description, when either the cluster programs or the party switchover sections  
10 of the living and standby parties are so described as to communicate with each other, a process similar to the communication 0781 is carried out even if not noted specifically.

To add, the information such as replica

15 status flag acquired from the other party in the step 0704 is stored in the replica status management table (Fig. 16) of the self-party. For example, when the standby party acquires from the living party a living party status flag "B2" the living party holds in  
20 relation to a pair indicated by a pair identifier of "2", the value "B2" is stored at a row corresponding to the pair identifier "2" in replica status flag in the column 1603 of living party status flag held by the standby party.

25 Subsequently, the party switchover section  
125 decides whether the copy VOL information reflection is necessary (step 0705). The reason for this is as below. In the normal state, the living party carries

out the copy VOL information change process (steps 0801 to 0804 in Fig. 8) and the standby party carries out the copy VOL information reflection process (steps 0921 to 0929 in Fig. 9) but for example, after the party 5 switchover due to a fault in the living party is done, there is a possibility that in the former standby party now acting as the living party, the party switchover has been done without performing the copy VOL information reflection process. Accordingly, in the 10 step 0705, it is decided whether the status flag of each party is "0" and if the status flag of any one of the parties is not "0", implying that the party 15 switchover is done during the copy VOL change (or during copy VOL information reflection process), the copy VOL information reflection process is determined to be necessary.

In case the copy VOL information reflection process is determined to be necessary in the step 0705, the cluster program 120 executes a copy VOL reflection 20 process in the fault recovery mode (11A in Fig. 11). If unnecessary, the cluster program 120 decides, in accordance with the presence or absence of the VOL replica execution request confirmed in the step 0782, whether the VOL replica needs to be executed (step 25 0706). When the step 0706 determines the necessity of execution, the cluster program 120 executes the VOL replica execution process (8A in Fig. 8). If unnecessary, the cluster program 120 again returns to

the self-party check process 0701 to continue the process.

Next, the cluster program 220 on standby party 200 first checks the status of self-party 5 similarly to the step 0701 (step 0721, arrow 0784) to decide whether the self-party is normal (step 0722). If the step 0722 determines the self-party to be abnormal, the party switchover section 225 on the cluster program 220 stops and ends the monitor process 10 (step 0723). On the other hand, if the self-party is normal, the party switchover section 225 communicates with the living party 100 through the communication 0781 (step 0724). Subsequently, the cluster program 220 decides whether the copy VOL information reflection 15 is necessary (step 0725). This step is provided for the same reason as that of the provision of the step 0705 and it is decided in the step 0725 similarly to the step 0705 whether the status flag of each party is "0" and if the status flag of any one of the parties is 20 not "0", implying that the party switchover has been done during the copy VOL change (or during the copy VOL information reflection process), the copy VOL information reflection process is determined to be necessary.

25 When the copy VOL information reflection is necessary, the cluster program 220 executes the copy VOL reflection process in the fault recovery mode (12B in Fig. 12). If unnecessary, the party switchover

section 225 decides, in accordance with the status of living party 100 acquired in the communication 0781, whether the party switchover is needed (step 0726). In case the party switchover takes place, the cluster  
5 program 220 informs the application 210 of it (arrow 0785) and the party switches to the living party (step 0727). Further, the cluster program 220 executes the process of monitoring the living party (7A in Fig. 7). On the other hand, if the party switchover is  
10 unnecessary, the cluster program 220 decides, in accordance with the presence or absence of the execution of VOL replica in the living party acquired through the communication 0781, whether the VOL replica is executed in the living party (step 0728). If the  
15 execution is to be done, the cluster program 220 executes the VOL execution process (8B in Fig. 8) but if unnecessary, the process again returns to the step 0721 to continue.

Illustrated in Figs. 8, 9 and 10 are  
20 flowcharts showing details of the copy VOL information consistency guaranty process shown in Figs. 4 and 5, with Fig. 8 depicting the stage X (copy VOL change process), Fig. 9 depicting the stage Y (copy VOL change reflection process) and Fig. 10 depicting the stage Z  
25 (copy VOL working resumption process).

In Fig. 8, before executing the VOL replica, the living party 100 informs the standby party of the start of copy VOL change (step 0801, arrow 0881) and

the standby party 200 receives this notice (step 0821). After the step 0821, the standby party 200 sets a status flag B1 indicating that the VOL replica process is in execution (step 0822) and executes the copy VOL  
5 status reflection process (9B in Fig. 9). Here, the flag setting process 0822 includes a step in which the party switchover section 225 stores the flag in the VOL replica file 240 (arrow 0884). In the following description, a flag storing process (a storage process)  
10 similar to that 0884 will be carried out in the flag setting process even when not mentioned specifically.

On the other hand, after the step 0801, the cluster program 120 on the living party 100 sets a status flag A1 indicative of the VOL replica execution  
15 start (step 0802) and then executes VOL replica (step 0803). When the VOL replica step 0803 ends, the cluster program 120 sets a status flag A2 indicative of the execution completion (step 0804, arrow 0883) and executes the copy VOL status reflection process (9A in  
20 Fig. 9).

Here, the steps 0801 and 0802 correspond to the step 0404 or 0504, the steps 0803 and 0804 correspond to the step 0405 or 0505, the steps 0821 and 0822 correspond to the step 0424 or 0524 and the  
25 communication 0881 corresponds to the communication 0481 or 0581.

In Fig. 9, the cluster program 120 of living party 100 first decides whether pair division is

carried out in the VOL replica process (step 0901). The VOL replica process will be executed in the step 0803 or step 1104 to be described later. The step 0901 is carried out for the following reason. During pair division, the standby party accesses a copy VOL and hence the copy VOL needs to be released whereas during pair reconstruction, the copy VOL need not be released to permit the living party to start working of the copy VOL. Accordingly, in the pair division mode, the cluster program 120 releases the copy VOL (step 0902). Thereafter, the cluster program 120 informs the standby party of completion of the copy VOL change (step 0903, arrow 0981) and after setting a status flag A3 indicating that the standby party 200 is executing the copy VOL status reflection process (step 0904), the living party executes the copy VOL working resumption process (10A in Fig. 10). It is to be noted that together with the notice of copy VOL change completion (step 0902) (or at a different timing), information indicating whether the replica targeting the copy VOL is pair division or pair reconstruction may be transmitted to the standby party 200. Here, the notice of copy VOL change completion or the aforementioned transmission of the information indicative of either the pair division or the pair reconstruction to the standby party 200 can be made, when the living party 100 detects the execution of the replica targeting the copy VOL or when the copy VOL lock release 0902 is

completed.

On the other hand, when receiving the communication 0981 (step 0921), the cluster program 220 on the standby party 200 sets a status flag B2 5 indicative of the execution start of the copy VOL change reflection process (step 0922). Thereafter, as in the step 0901, it is decided whether the pair division is carried out (step 0923). The reason for this is as below. During the pair division, the 10 standby party accesses the copy VOL and therefore lock control must be done whereas during the pair construction, the copy VOL information is merely erased and access to the copy VOL is unnecessary. It is to be noted that a decision as to whether the pair division 15 is carried out can be made by directly consulting the information 2 transmitted from the living party computer or by acquiring the information 2 correspond- ing to a physical (device) name of the copy VOL subjected to the copy volume reflection process by 20 making reference to the replica status management table storing the VOL replica status flag. On the other hand, in 1109 of Fig. 11 and 1205 of Fig. 12, a decision is made by the latter method, that is, by making reference to the information 2 stored in the 25 replica status management table in correspondence with the physical (device) name of the copy VOL subjected to the copy volume reflection process.

When the pair division is determined, the

cluster program 220 acquires a PVID of the copy VOL (step 0925) and thereafter, obtains the copy VOL lock (step 0926). Then, it acquires the copy VOL information (step 0927) and releases the copy VOL lock 5 (step 0928). On the other hand, when the pair reconstruction is determined, the cluster program 120 erases the copy VOL information including the PVID of the copy VOL (step 0924). Through this, reflection of the copy VOL information upon the standby party 200 is 10 completed in the respective modes. Thereafter, a status flag B3 indicative of the reflection completion is set (step 0929) and the copy VOL working resumption process (10B in Fig. 10) is executed.

Here, the step 0902 corresponds to the step 15 0406, the steps 0903 and 0904 correspond to the step 0407 or 0506, the steps 0921 and 0922 correspond to the step 0425 or 0525, the step 0924 corresponds to the step 0526 and the communication 0981 corresponds to the communication 0484 or 0583. Further, the steps 0925 to 20 0928 correspond to the steps 0426 to 0429 in sequence.

In Fig. 10, the cluster program 220 on standby party 200 informs the cluster program 120 on living party 100 that the copy VOL information reflection is ended (step 1021, arrow 1081). After the 25 step 1021, the cluster program 220 sets a status flag "0" indicating that the information has been reflected (step 1022) and the process again returns to the fault monitor process (7B in Fig. 7) to continue. On the

other hand, when the cluster program 120 receives the communication 1081 (step 1001), it sets a status flag "0" indicating that the information has been reflected (step 1082). Thereafter, the cluster program 120

5 decides whether the pair division is done similarly to the step 0901 (step 1003). The reason for this is as below. During the pair division, the copy VOL is released in the stage Y and the copy VOL must be reacquired whereas in the pair division mode, the

10 living party has started working of the copy VOL in the stage Y. Accordingly, only in the pair division mode, the cluster program 120 executes acquisition of the copy VOL lock (step 1004) and thereafter informs the application 110 that working of the copy VOL is

15 permissible (step 1005, arrow 1083). Subsequently, the process again returns to the fault monitor process (7A in Fig. 7) to continue.

Here, the steps 1001 and 1002 correspond to the step 0408 or 0507, the step 1004 corresponds to the

20 step 0409, the steps 1021 and 1022 correspond to the step 0430 or 0527, and the communication 1081 corresponds to the communication 0489 or 0583.

Further, the step 1005 is included in the step 0410.

Illustrated in Fig. 11 is a flowchart showing

25 the procedures for the living party to take over the VOL replica process and copy VOL in the event that a fault takes place. For steps in the process of Fig. 11 similar to those described in connection with Figs. 8,

9 and 10, only the correspondence is indicated for simplicity of explanation. Firstly, the cluster program 120 of living party 100 first consults the VOL replica status flag 222 to decide whether the status 5 flag is "0" (step 1101). If the flag is "0" in the step 1101, it is indicated that the copy VOL information is reflected correctly. Therefore, in this case, the cluster program 120 does not proceed with the present process but simply executes the copy VOL 10 information reflection process (12A, Fig. 12) upon the standby party to be done in the event of the occurrence of a fault. If, in the step 1101, the flag is not "0", it is then decided whether the VOL replica has been completed during the fault occurrence (step 1102). In 15 case the VOL replica has not been completed, the following instances will prevail, including an instance in which the living party serving as the former living party has the status flag A1, an instance in which the living party serving as the former standby party has 20 the status flag B1 and the standby party serving as the former living party has the status flag A1. Here, the status flag of the standby party indicates the status flag of standby party acquired in the step 0704 in Fig. 7 which includes process to call the process 11A.

25 When, in the step 1102, the status flag of living party is not A1, or the status flag of living party is B1 and the status flag of the other party is not A2 or A3, it is indicated that the VOL replica is

in execution and the living party has not yet completed the change and therefore, the cluster program 120 executes the VOL replica corresponding to the stage X (Fig. 4) to reflect the copy VOL information upon the 5 living party (steps 1103 to 1105). Here, the steps 1103 to 1105 correspond to the steps 0802 to 0804, respectively. After a series of the reflection steps, the copy VOL information reflection process (12A in Fig. 12) upon the standby party is executed.

10 On the other hand, when in the step 1102 "N" is determined, it is indicated that the VOL replica has already been executed. Then, the process is carried out in accordance with how far the standby party reflection process after the execution is proceeded 15 with.

Firstly, the cluster program 120 decides whether the status flag is A2 or A3 (step 1106). In case A2 or A3 is determined in the step 1106, it is indicated that the copy VOL information in the living 20 party has been reflected and the copy VOL information reflection completion notice from the standby party is waited for and therefore, the cluster program 120 releases the notice wait state (step 1107). Next, when neither A nor A3 is determined in the step 1106, it is 25 decided whether the status flag is B2 (step 1108). If in the step 1108 the status flag is B2, implying that the standby party in the information reflection process is switched to the living party, the cluster program

120 continues the information reflection process to reflect the copy VOL information. In the reflection process, like the stage Y (7B in Fig. 7), different steps are carried out in accordance with either the  
5 pair division mode or the pair reconstruction mode (steps 1109 to 1114). Here, the steps 1109 to 1114 correspond to the steps 0923 to 0928, respectively. On the other hand, if in the step 1108 the flag is not B2, it is indicated that the status flag is B3 and the  
10 standby party having completed the reflection of the copy VOL status is switched to the living party and therefore the copy VOL information reflection has been completed and no step needs to be undertaken.

Through the above, even when the status flag  
15 is neither A1 nor B1 in the step 1102, take-over of the copy VOL information has been completed and therefore, the status flag is set to "0" (step 1115) and like the stage Z of living party (Fig. 10), the copy VOL working start process is carried out in accordance with either  
20 the pair division or the pair reconstruction mode (steps 1116 to 1118, arrow 1181). Here, the steps 1116 to 1118 correspond to the steps 1003 to 1005, respectively and the communication 1181 corresponds to the communication 1083. Thereafter, the copy VOL  
25 information reflection process upon the standby party (12A in Fig. 12) is carried out.

Illustrated in Fig. 12 is a flowchart showing the procedures for the volume information taken over to

the living party computer in the event that the living/standby party computers become faulty is taken over to the standby party computer. Like Fig. 11, for steps in Fig. 12 similar to those described in  
5 connection with Figs. 8, 9 and 10, only the correspondence will be described for simplicity of explanation.

Firstly, each of the cluster programs 120 and 220 of the living and standby parties 100 and 200  
10 mutually transmits the status flag of the self-party to the other party to recognize the party status mutually (steps 1201 1221, arrow 1281). The reasons for this are as below: that is, for deciding whether the copy VOL status reflection process of the standby party has  
15 already been reflected in accordance with the status flag of the standby party (the standby party status flag being "0" or A2, or A3 or B3), for deciding an instance in which when the copy VOL status reflection is necessary in the standby party, the standby party  
20 needs to execute the stage X (the living party status flag being A1 or B1) in accordance with the status flag of the living party, and because the process is different for the case where the standby party executes the copy VOL take-over process (stage Y) and for the  
25 case where the standby party reflecting the copy VOL status is switched and the living party needs to take over the copy VOL status (the living party status flag being B2).

Accordingly, the cluster programs 120 and 220 decide whether the standby party status flag is 0 or A2, or A3 or B3 (steps 1202 and 1222). When in the steps 1202 and 1222, "Y" is issued, the living and 5 standby parties return to the fault monitor process to continue the process (7A and 7B in Fig. 7). On the other hand, when in the steps 1202 and 1222, "N" is issued, it is indicated that the standby party needs to reflect the copy VOL status and therefore, the cluster 10 programs 120 and 220 decide whether the living party status flag is A1 or B1 (steps 1203 and 1223). If, in the steps 1203 and 1222, A1 or B1 is settled, the living and standby parties carry out the execution starting from the stage X (8A/8B in Fig. 8).

15           When the results in the steps 1203 and 1223 are other than the above, the standby party 200 must execute the VOL reflection process and therefore it executes the stage Y (9B in Fig. 9). On the other hand, the living party 100 decides by means of the 20 cluster program 120 whether the living party status flag is B2 (step 1204). When in the step 1204 the flag is B2, implying that the copy VOL status needs to be taken over, a step similar to the stage Y of standby party (9B in Fig. 9) is carried out to reflect the copy 25 VOL information upon the living party (steps 1205 to 1208). Here, the steps 1205 to 1207 correspond to the steps 0923 to 0925 and the step 1208 corresponds to the step 0927. When the steps 1205 to 1208 are completed,

causing the living party to have completed the information reflection, the status flag A2 is set through a step similar to the step 0804 (step 1209). Then, the stage Y is executed (9A in Fig. 9). On the 5 other hand, the flag is determined not to be B2 in the step 1204, the stage Y is executed as it is (9A in Fig. 9).

Advantageously, through the processes shown in Figs. 11 and 12, even when a fault occurs in the 10 course of the execution of the consistency guaranty process of copy VOL information shown in Figs. 6 to 10, the living party continuously takes over the process in execution to guarantee the consistency of the copy VOL information. In addition, the above advantage can be 15 combined with an advantage that the party switchover can be guaranteed when a fault occurs after the consistency guaranty process shown in Figs. 4 and 5 to bring about an advantage that a highly utilizable system can be constructed which can take over the 20 process including the VOL replica from the living party to the standby party even in the event that a fault occurs in the living party/standby party computer system executing the VOL replica. The following description will be given by way of example of 25 operation in which after the communication process 0484 in the stage Y of Fig. 4 (step 0407), a fault occurs in the living party and the party switchover is effected. In this case, the copy VOL change process is in the

pair division mode and is completed and therefore, the party switchover is done while the status flag A3 being set in the living party and the status flag B2 being set in the standby party to execute the flowchart shown 5 in Fig. 7. After the party switchover, the living party serving as the former standby party has the status flag B2 whereas the standby party serving as the former living party has the status flag A3.

Firstly, in Fig. 7, the copy VOL information 10 reflection process is determined to be necessary in the steps 0705 and 0725. Through this, the living party executes the process indicated at 11A in Fig. 11 and the standby party executes the process indicated at 12B in Fig. 12.

15 Subsequently, the living party holding the status flag B2 in Fig. 11 proceeds to the step 1108 through the steps 1101, 1102 and 1106. In the step 1108, "Y" is issued. This implies that the copy VOL change process has already been completed in the former 20 living party and because of the pair division mode, the living party serving as the former standby party performs the copy VOL information reflection process (steps 1111 to 1113). Because of the completion of the copy VOL information reflection process, a copy VOL is 25 created through pair division after the status flag is cleared and therefore, this copy VOL is used to start copy VOL working (steps 1116 to 1118). Thereafter, the process shifts to 12A in Fig. 12.

In Fig. 12, as described previously, the living party having the status flag "0" first executes the step 1201 and the standby party having the status flag A3 executes the step 1221. As a result, because 5 of the standby party status flag being A3, the copy VOL information reflection is determined as being completed and both the living and standby parties shift to the process indicated at 7A and 7B in Fig. 7 representing the normal working state.

10 Thus, when a fault occurs in the living party after the communication process 0484 in the stage Y of Fig. 4 is done, the copy VOL information reflection process is continuously proceeded with even after the party switchover, thereby ensuring that the consistency 15 of the copy VOL information can be kept and the living party can start working by using the copy VOL.

In the foregoing description, another embodiment has been described in which only the copy VOL completion notice is effected before the party 20 switchover and the copy VOL information reflection process is carried out after the party switchover takes place.

Thereafter, even when a fault occurs in the living party, the standby party is permitted to access 25 the copy VOL so as to normally take over both the original and copy VOL service affairs, thereby solving the conventional problems.

Although in the present embodiment the fault

has been described as being caused in the living and standby parties, the technique of the present embodiment can be applied to an instance in which a fault occurs in the network utilized by the  
5 communication means 01, by taking the priority of the living and standby parties into consideration.

As described above, the changed copy volume information is reflected upon the standby party by taking the opportunity of the execution of the volume  
10 replica causing the copy volume to change and therefore, even when a fault occurs in the living party after the execution of the volume replica, the process can be handed over to the standby party. Further, even when a fault occurs during the execution of the volume  
15 replica and copy volume reflection process, the process being in execution at the time that the fault occurs can be taken over after the party switchover.

As has been set forth so far, according to the present invention, the change of the copy VOL due  
20 to the VOL replica means can be reflected upon the standby party.